Amendments to the Claims

The listing of claims will replace all prior versions, and listings of claims in the application.

1. (Previously amended) A multiply unit comprising:

at least one input data path for receiving one or more input operands to the multiply unit;

an arithmetic multiplier connected to receive the one or more input operands;

a binary polynomial multiplier connected to receive the one or more input operands and including components separate and distinct from components of the arithmetic multiplier;

permutation logic connected to receive the one or more input operands and operable to produce an output comprising a permutation of the one or more operands; and

a multiply unit output data path connected to receive an output of the arithmetic multiplier, connected to receive an output of the binary polynomial multiplier, and connected to receive the output of the permutation logic,

wherein the multiply unit output data path includes one or more components to separately select the output of the arithmetic multiplier, the output of the binary polynomial multiplier or the output of the permutation logic to form a result.

2. (Original) The multiply unit of claim 1, wherein the arithmetic multiplier includes a multiplier array.

3. (Original) The multiply unit of claim 2, wherein the multiplier array is a Wallace tree multiplier array.

4. (Original) The multiply unit of claim 2, wherein the multiplier array includes a plurality of carry-save adders arranged in a tree structure.

5. (Original) The multiply unit of claim 4, further comprising a carry-propagate adder.

6. (Original) The multiply unit of claim 1, further comprising Booth recoding logic.

7. (Previously Presented) The multiply unit of claim 2, wherein the arithmetic multiplier performs 32-bit by 16-bit multiplications in two clock cycles.

8. (Original) The multiply unit of claim 2, wherein the arithmetic multiplier performs 32-bit by 32-bit multiplications in three clock cycles.

9. (Original) The multiply unit of claim 1, wherein the binary polynomial multiplier includes a binary polynomial multiplication array.

10. (Original) The multiply unit of claim 9, wherein the binary polynomial multiplier includes a polynomial multiplication array having a first input and a second input, the polynomial multiplication array including:

a plurality of row multipliers that multiply the first input by a bit of the second input; and

at least one adder for computing a result by adding the results from the plurality of row multipliers.

11. (Original) The multiply unit of claim 10, wherein the at least one adder performs a bitwise exclusive-or on the results from the plurality of row multipliers.

12. (Original) The multiply unit of claim 10, wherein at least one of the plurality of row multipliers performs multiplication by computing a logical AND of the first input and a bit of the second input.

13. (Original) The multiply unit of claim 10 further comprising an accumulator, and wherein the at least one adder computes a result by adding the results from the plurality of row multipliers and the accumulator.

14. (Cancelled).

15. (Previously amended) In a processor core, a method for performing polynomial arithmetic, the method comprising:

fetching an instruction to perform an operation from a data store;

reading one or more registers;

performing the operation using a multiply unit, the multiply unit comprising:

at least one input data path for receiving one or more input operands to the multiply unit;

an arithmetic multiplier connected to receive the one or more input operands;

a binary polynomial multiplier connected to receive the one or more input operands and including components separate and distinct from components of the arithmetic multiplier;

permutation logic connected to receive the one or more input operands and operable to produce an output comprising a permutation of the one or more input operands; and

a multiply unit output data path connected to receive an output of the arithmetic multiplier, connected to receive an output of the binary polynomial multiplier, and connected to receive the output of the permutation logic,

wherein the multiply unit output data path includes one or more components to separately select the output of the arithmetic multiplier, the output of the binary polynomial multiplier or the output of the permutation logic to form a result.

16. (Original) The method of claim 15, wherein the arithmetic multiplier includes a multiplier array.

17. (Original) The method of claim 16, wherein the multiplier array is a Wallace tree multiplier array.

18. (Original) The method of claim 16, wherein the multiplier array includes a plurality of carry-save adders arranged in a tree structure.

19. (Original) The method of claim 18, the multiply unit further comprises a carry-propagate adder.

20. (Original) The method of claim 15, further comprising Booth recoding logic.

21. (Previously Presented) The method of claim 16, wherein the arithmetic multiplier performs 32-bit by 16-bit multiplications in two clock cycles.

22. (Original) The method of claim 16, wherein the arithmetic multiplier performs 32-bit by 32-bit multiplications in three clock cycles.

23. (Original) The method of claim 15, wherein the binary polynomial multiplier includes a binary polynomial multiplication array.

24. (Original) The method of claim 23, wherein the binary polynomial multiplier includes a polynomial multiplication array having a first input and a second input, the polynomial multiplication array including:

a plurality of row multipliers that multiply the first input by a bit of the second input; and

at least one adder for computing a result by adding the results from the plurality of row multipliers.

25. (Original) The method of claim 24, wherein the at least one adder performs a bitwise exclusive-or on the results from the plurality of row multipliers.

26. (Original) The method of claim 24, wherein at least one of the plurality of row multipliers performs multiplication by computing a logical AND of the first input and a bit of the second input.

27. (Original) The method of claim 24, wherein the multiply unit further comprises an accumulator, and wherein the at least one adder computes a result by adding the results from the plurality of row multipliers and the accumulator.

28. (Cancelled).

29. (Previously amended) A computer-readable medium comprising a microprocessor core embodied in software, the microprocessor core including a multiply-divide unit, the multiply-divide unit comprising:

at least one input data path for receiving one or more input operands to the multiply unit;

an arithmetic multiplier connected to receive the one or more input operands;

a binary polynomial multiplier connected to receive the one or more input operands and including components separate and distinct from components of the arithmetic multiplier;

permutation logic connected to receive the one or more input operands and operable to produce an output comprising a permutation of the one or more operands; and

a multiply unit output data path connected to receive an output of the arithmetic multiplier, connected to receive an output of the binary polynomial multiplier, and connected to receive the output of the permutation logic,

wherein the multiply unit output data path includes one or more components to separately select the output of the arithmetic multiplier, the output of the binary polynomial multiplier or the output of the permutation logic to form a result.

30. (Original) The computer-readable medium of claim 29, wherein the arithmetic multiplier includes a multiplier array.

31. (Original) The computer-readable medium of claim 30, wherein the multiplier array is a Wallace tree multiplier array.

32. (Original) The computer-readable medium of claim 30, wherein the multiplier array includes a plurality of carry-save adders arranged in a tree structure.

33. (Original) The computer-readable medium of claim 32, wherein the multiply unit further comprises a carry-propagate adder.

34. (Original) The computer-readable medium of claim 29, further comprising Booth recoding logic.

35. (Previously Presented) The computer-readable medium of claim 30, wherein the arithmetic multiplier performs 32-bit by 16-bit multiplications in two clock cycles.

36. (Original) The computer-readable medium of claim 30, wherein the arithmetic multiplier performs 32-bit by 32-bit multiplications in three clock cycles.

37. (Original) The computer-readable medium of claim 29, wherein the binary polynomial multiplier includes a binary polynomial multiplication array.

38. (Original) The computer-readable medium of claim 37, wherein the binary polynomial multiplier includes a polynomial multiplication array having a first input and a second input, the polynomial multiplication array including:

a plurality of row multipliers that multiply the first input by a bit of the second input; and

at least one adder for computing a result by adding the results from the plurality of row multipliers.

39. (Original) The computer-readable medium of claim 38, wherein the at least one adder performs a bitwise exclusive-or on the results from the plurality of row multipliers.

40. (Original) The computer-readable medium of claim 38, wherein at least one of the plurality of row multipliers performs multiplication by computing a logical AND first input and a bit of the second input.

41. (Original) The computer-readable medium of claim 38, wherein the multiply unit further comprises an accumulator, and wherein the at least one adder

computes a result by adding the results from the plurality of row multipliers and the accumulator.

42. (Cancelled).